

1 Настройка триггера

Обзор

Для настройки триггера, сделайте следующее:

- Перейдите в: *Настройка* → *Узлы сети*
- Нажмите на *Триггеры* в строке с узлом сети
- Нажмите на *Создать триггер* сверху справа (или на имя триггера для изменения уже существующего триггера)
- Введите в диалоге параметры триггера

Настройка

Вкладка **Триггер** содержит все существенные атрибуты триггера.

Trigger
Dependencies

* Name

Severity Not classified Information Warning Average High

* Expression Add

[Expression constructor](#)

OK event generation Expression Recovery expression None

PROBLEM event generation mode Single Multiple

OK event closes All problems All problems if tag values match

Tags

tag	value
-----	-------

[Add](#)

Allow manual close

URL

Description

Enabled

Add
Cancel

Все обязательные поля ввода отмечены красной звёздочкой.

Параметр	Описание
Имя	<p>Имя триггера.</p> <p>Имя может содержать поддерживаемые макросы: {HOST.HOST}, {HOST.NAME}, {HOST.CONN}, {HOST.DNS}, {HOST.IP}, {ITEM.VALUE}, {ITEM.LASTVALUE} и {\$MACRO}.</p> <p>Можно использовать макросы \$1, \$2...\$9 как ссылки на первую, вторую... девятую константу из выражения.</p> <p><i>Обратите внимание:</i> Макросы \$1-\$9 будут раскрыты корректно, если константы на которые ссылаются макросы, в относительно простых, понятных выражениях. Например, имя "Загрузка процессора превышает \$1 на {HOST.NAME}" будет автоматически изменено на "Загрузка процессора превышает 5 на New host", если выражение - {New host:system.cpu.load[percpu,avg1].last()}>5</p>

Параметр	Описание
Важность	Определение требуемой важности триггера нажатием кнопок.
Выражение проблемы	Логическое выражение , используемое для определения условий проблемы.
Формирование ОК событий	<p>Опции формирования ОК событий:</p> <p>Выражение - ОК события формируются на основе того же выражения что и события о проблемах;</p> <p>Выражение восстановления - ОК события формируются, если выражение проблемы вычисляется как ЛОЖЬ и выражение восстановления вычисляется как ПРАВДА;</p> <p>Нет - в этом случае триггер никогда самостоятельно не вернётся в состояние ОК.</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p>
Выражение восстановления	<p>Логическое выражение, используемое для определения условий, когда проблема решена.</p> <p>Выражение восстановления вычисляется только после того, как выражение проблемы будет вычислено как ЛОЖЬ. Невозможно решить проблему с помощью выражения восстановления, если условие проблемы всё ещё присутствует.</p> <p>Это поле опционально и доступно только, если в <i>Формирование ОК событий</i> выбрано 'Выражение восстановления'.</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p>
Режим формирования ПРОБЛЕМА событий	<p>Режим формирования событий о проблемах:</p> <p>Одиночный - формируется одно событие, когда триггер переходит в состояние 'Проблема' в первый раз;</p> <p>Множественный - событие формируется на <i>каждое</i> 'Проблема' вычисление триггера.</p>
ОК событие закрывает	<p>Выберите, если ОК событие закрывает:</p> <p>Все проблемы - все проблемы этого триггера</p> <p>Все проблемы, если значение тега совпадает - только те проблемы триггера, значения тегов событий которых совпадает</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p>
Теги для совпадения	<p>Укажите имя тега события, которое будет использоваться для корреляции событий.</p> <p>Это поле отображается, если в свойстве <i>ОК событие закрывает</i> выбрано 'Все проблемы, если значения тегов совпадают' и в этом случае это поле обязательно.</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p>
Теги	<p>Укажите пользовательские теги для маркировки событий триггера.</p> <p>Теги событий могут использоваться для корреляции событий, в условиях действий и будут также видны в <i>Мониторинг → Проблемы</i> и в <i>Проблемы</i> виджете.</p> <p>Тегами является пара имя тега и значение. Вы можете использовать только имя или на пару имя со значением.</p> <p>Пользовательские макросы, контекст пользовательских макросов, макросы низкоуровневого обнаружения и макросы функций <code>{{ITEM.VALUE}}</code>, <code>{{ITEM.LASTVALUE}}</code>, а также макросы низкоуровневого обнаружения, поддерживаются в тегах событий.</p> <p>Макросы низкоуровневого обнаружения можно использовать в контексте макросов.</p> <p>Если суммарная длина раскрытого значения превышает, оно будет обрезано до 255 символов.</p> <p>Смотрите все поддерживаемые макросы тегами событий.</p> <p>Поддерживается начиная с Zabbix 3.2.0.</p>

Параметр	Описание
<i>Разрешить закрывать вручную</i>	Выберите, чтобы разрешить закрывать вручную события о проблемах сформированные этим триггером. Закрытие вручную возможно при подтверждении событий о проблемах. Поддерживаются начиная с Zabbix 3.2.0.
<i>URL</i>	Если не пустой, то URL, указанный здесь, доступен в виде ссылки в нескольких местах веб-интерфейса, например при нажатии на имя проблемы в <i>Мониторинг</i> → <i>Проблемы</i> (опция <i>URL</i> в меню <i>Триггер</i>) и в виджете <i>Проблемы</i> на панели. Поддерживаемые макросы: {ITEM.VALUE}, {ITEM.LASTVALUE}, {TRIGGER.ID}, несколько {HOST.*} макросов, пользовательские макросы.
<i>Описание</i>	Текстовое поле используется, для того чтобы сообщить больше информации об этом триггере. Может содержать инструкции по устранению специфичной проблемы, контактные данные ответственного сотрудника и т.п. <i>Начиная с Zabbix 2.2</i> , описание может содержать тот же набор макросов, что и имя триггера.
<i>Активировано</i>	Уберите отметку деактивирует триггер, если это требуется.

Вкладка **Зависимости** содержит все **зависимости** триггера.

Нажмите на *Добавить* для добавления новой зависимости.

Вы также можете настроить триггер, открыв уже существующий триггер, затем нажать на кнопку *Клонировать* и сохранить его под другим именем.

Тестирование выражений

Имеется возможность проверки настроенного выражения триггера относительно того, как результат выражения будет меняться в зависимости от полученного значения.

Следующее выражение с официального шаблона взято в качестве примера:

```
{Template Net Cisco IOS
SNMPv2:sensor.temp.value[ciscoEnvMonTemperatureValue.{#SNMPINDEX}].avg(5m)}>
{$TEMP_WARN}
or
{Template Net Cisco IOS
SNMPv2:sensor.temp.status[ciscoEnvMonTemperatureState.{#SNMPINDEX}].last(0)}
={$TEMP_WARN_STATUS}
```

Для тестирования выражения, нажмите на *Конструктор выражения* под полем выражения.

Trigger Dependencies

* Name

Severity Not classified Information Warning Average High

* Expression

```
{Template Net Cisco IOS
SNMPv2:sensor.temp.value[ciscoEnvMonTemperatureValue.
{#SNMPINDEX}].avg(5m)}>{$TEMP_WARN}
or
{Template Net Cisco IOS
SNMPv2:sensor.temp.status[ciscoEnvMonTemperatureState.
{#SNMPINDEX}].last(0)}={$TEMP_WARN_STATUS}
```

[Expression constructor](#)

В Конструкторе выражения перечислены все отдельные выражения. Чтобы открыть окно тестирования, нажмите на *Тест* ниже списка выражений.

Target Expression

Or Action Info

A {Template Net Cisco IOS SNMPv2:sensor.temp.value[ciscoEnvMonTemperatureValue.{#SNMPINDEX}].avg(5m)}>{\$TEMP_WARN} [Remove](#)

B {Template Net Cisco IOS SNMPv2:sensor.temp.status[ciscoEnvMonTemperatureState.{#SNMPINDEX}].last(0)}={\$TEMP_WARN_STATUS} [Remove](#)

[Test](#)

В окне тестирования вы можете задать пробные значения (“80, 70, 0, 1” в этом примере) и затем увидеть результат выражения, нажав на кнопку *Тест*.

Test

Expression Variable Elements	Result type	Value
{Template Net Cisco IOS SNMPv2:sensor.temp.value[ciscoEnvMonTemperatureValue.{#SNMPINDEX}].avg(5m)}	Numeric (float)	<input type="text" value="80"/>
{\$TEMP_WARN}	Numeric (float)	<input type="text" value="70"/>
{Template Net Cisco IOS SNMPv2:sensor.temp.status[ciscoEnvMonTemperatureState.{#SNMPINDEX}].last(0)}	Numeric (integer)	<input type="text" value="0"/>
{\$TEMP_WARN_STATUS}	Numeric (float)	<input type="text" value="1"/>

Expression	Result
Or	TRUE
A {Template Net Cisco IOS SNMPv2:sensor.temp.value[ciscoEnvMonTemperatureValue.{#SNMPINDEX}].avg(5m)}>{\$TEMP_WARN}	TRUE
B {Template Net Cisco IOS SNMPv2:sensor.temp.status[ciscoEnvMonTemperatureState.{#SNMPINDEX}].last(0)}={\$TEMP_WARN_STATUS}	FALSE
A or B	TRUE

[Test](#) [Cancel](#)

Можно увидеть результат отдельных выражений, наряду со полным выражением.

Результат “ПРАВДА” означает, что введёное выражение корректно. В этом конкретном выражении A, “80” больше чем указанное значение { \$TEMP_WARN }, “70” в этом примере. Соответственно, появится результат “ПРАВДА”.

Результат “ЛОЖЬ” означает, что введёное выражение некорректно. В этом конкретном выражении B, { \$TEMP_WARN_STATUS }, равно “1” в этом примере, должно быть равно указанному значению 0, и это неправда. Соответственно, появится результат “ЛОЖЬ”.

Выбранным типом выражения является “ИЛИ”/“ПРАВДА”. Если по крайней мере одно из

заданных выражений (А или В в этом случае) будет ПРАВДА, общий результат будет также ПРАВДА. Что означает, что текущее значение превысило пороговое значение и возникнет Проблема.

From:

<https://www.zabbix.com/documentation/5.2/> - **Zabbix Documentation 5.2**

Permanent link:

<https://www.zabbix.com/documentation/5.2/ru/manual/config/triggers/trigger>

Last update: **2019/10/07 06:35**

